

UNOFFICIAL COMMUNICATION FOR EXAMINER REVIEW ONLY – PLEASE DO NOT ENTER

LAW OFFICES

**AMIN, TUROCY & CALVIN, LLP**57<sup>TH</sup> FLOOR, KEY TOWER

127 PUBLIC SQUARE

CLEVELAND, OHIO 44114

TELEPHONE: 216-696-8730

FACSIMILE: 216-696-8731

EMAIL: TVANTUINEN@THEPATENTATTORNEYS.COM

**FACSIMILE TRANSMISSION**

---

THIS MESSAGE IS INTENDED ONLY FOR THE USE OF THE INDIVIDUAL OR ENTITY TO WHICH IT IS ADDRESSED AND MAY CONTAIN INFORMATION THAT IS PRIVILEGED, CONFIDENTIAL AND EXEMPT FROM DISCLOSURE UNDER APPLICABLE LAW. IF THE READER OF THIS MESSAGE IS NOT THE INTENDED RECIPIENT OR THE EMPLOYEE OR AGENT RESPONSIBLE FOR DELIVERING THE MESSAGE TO THE INTENDED RECIPIENT, YOU ARE HEREBY NOTIFIED THAT ANY DISSEMINATION, DISTRIBUTION OR COPYING OF THIS COMMUNICATION IS STRICTLY PROHIBITED. IF YOU HAVE RECEIVED THIS COMMUNICATION IN ERROR, PLEASE NOTIFY US IMMEDIATELY, AND RETURN THE ORIGINAL MESSAGE TO US AT THE ADDRESS LISTED BELOW VIA UNITED STATES MAIL. THANK YOU.

---

**Date:** November 3, 2008**TO:** Thuy Chan Dao – United States Patent and Trademark Office**FAX NO.:** (571) 273-8570**FROM:** Timothy Van Tuinen

In re patent application of:

Applicant(s): Raymond W. McCollum, *et al.*

Examiner: Thuy Chan Dao

Serial No: 10/692,432

Art Unit: 2192

Filing Date: October 23, 2003

Title: USE OF ATTRIBUTION TO DESCRIBE MANAGEMENT INFORMATION

---

TOTAL NUMBER OF PAGES (INCLUDING THIS PAGE): 11

## UNOFFICIAL COMMUNICATION FOR EXAMINER REVIEW ONLY – PLEASE DO NOT ENTER

Dear Examiner Dao:

Thank you for the opportunity to discuss this matter. The following agenda is provided to aid our discussion.

- I. Introduction – examiner suggestions.
- II. Discussion of Rejection of Claims under 35 U.S.C. §101:
  - a. Review of amendments.
- III. Discussion of Rejection of Claims under 35 U.S.C. §102(e):
  - a. Claims 1-13, 15-30, and 33-36 stand rejected as being anticipated by Gibbons *et al.* (US 2004/0034853).
    - i. Discussion of the cited reference with respect to the subject claims:
      1. Claim 1 recites *an attribution component that facilitates attributing selected parts of code of the application or service with management information*. Gibbons *et al.* fails to anticipate or suggest this novel feature of the subject claim. Rather, Gibbons *et al.*, paragraphs [0095]-[0101], describes a Server phase where a packager prepares the XHTML application for use on one or more MT devices. A packager extracts ECMAScript code from the XHTML file and from any script files referenced by the XHTML file. The script code may be classified by the packager as either top-level code or event-handler code. The packager keeps the extracted top-level code in the same order as it appears in the original XHTML file in order to preserve the specified execution order. Event-handler code is extracted separately from the top-level code. The packager keeps the extracted event-handler code in order, and assigns a unique sequence number to each fragment. The packager embeds a special XML tag in place of each removed code fragment of event-handler code. The XML tag includes a string containing unique sequence number associated with the removed code fragment. This XML tag is used to locate the

## UNOFFICIAL COMMUNICATION FOR EXAMINER REVIEW ONLY – PLEASE DO NOT ENTER

appropriate code fragment during the execution of the application. The removed application code, which includes both top-level and event-handler code, is then compiled into bytecodes. Thereafter, constant table data is appended and linked to the compiled bytecodes. The stripped XHTML is then processed with the appropriate CSS parameters. The resulting XHTML file contains only the elements required for the specified target MT device. The result is two separate data groups, a processed XHTML/CSS group and a compiled bytecode group. These two groups are then combined into a single file. Gibbons *et al.* merely describes extracting parts of code to enable various MT devices to be compatible with various applications. In contrast, the subject claim recites attributing selected parts of code of the application with management information. Through this feature, the subject claim facilitates a developer to avoid writing and maintaining two separate pieces of code. Therefore, Gibbons *et al.* fails to recite novel aspects of the subject claim. Clarification of the Examiner's interpretation is respectfully requested.

2. As per claim 2, Gibbons *et al.* merely describes a server that is in communication with developers and users which contains a configuration management component 455, as well as other various components. The configuration MC 455 manages application data. Gibbons *et al.* fails to describe ***a software tool that is applied to the attributed code of the application or service to expose the management information.***

3. Claim 3 recites ***the management information is exposed and used to generate a manifest.*** Figure 5 and paragraphs [0117]-[0122] of Gibbons *et al.* merely describes the various components of a content submission by a developer. Further clarification of the Examiner's interpretation of Gibbons *et al.* is respectfully requested.

## UNOFFICIAL COMMUNICATION FOR EXAMINER REVIEW ONLY – PLEASE DO NOT ENTER

4. Discussion of the remaining claim rejections as they pertain to the Gibbons *et al.* reference.
- IV. Discussion of Rejection of Claims under **35 U.S.C. §103(a)**:
- a. Claims 14, 31-32, and 37 stand rejected as being unpatentable over Gibbons *et al.* in view of Gschwind *et al.*
    - i. Discussion of the cited references with respect to the subject claims.

Thank you once again for your time and consideration.

Best regards,  
Timothy Van Tuinen

UNOFFICIAL COMMUNICATION FOR EXAMINER REVIEW ONLY – PLEASE DO NOT ENTER

**PROPOSED CLAIM AMENDMENTS – DO NOT ENTER**

**Listing of Claims:**

1. (Currently Amended) A machine-implemented system, embodied in a computer-readable storage medium, that facilitates management of an application or service, comprising:
  - an application or service for installation on the system; and
  - an attribution component that facilitates attributing selected parts of code of the application or service with management information;where the system uses the management information to manage the installed application or service.
2. (Original) The system of claim 1, a software tool is applied to the attributed code of the application or service to expose the management information.
3. (Original) The system of claim 1, the management information is exposed and used to generate a manifest.
4. (Previously Presented) The system of claim 1, the attributed code of the application or service indicates at least one subset of types within one or more components of the application or service that should be exposed and how the subset of types should be identified.
5. (Original) The system of claim 1, the management information is exposed from at least one of an in-process provider and a decoupled provider.

## UNOFFICIAL COMMUNICATION FOR EXAMINER REVIEW ONLY – PLEASE DO NOT ENTER

6. (Original) The system of claim 1, the management information is exposed from a decoupled provider, which attributed code of the decoupled provider includes at least one of a register call at startup and an unregister call at shutdown.

7. (Original) The system of claim 1, a catalog is created of all available instrumentation data of the system, wherein the catalog is browsed and used to discover a particular instrumentation point.

8. (Original) The system of claim 1, at runtime, management information is retrieved by identifying the associated management information within a catalog of all management information of the system, and following the associated management information to the corresponding application or service.

9. (Original) The system of claim 8, for an in-process provider at runtime, the component associated with the management information, is loaded and invoked.

10. (Original) The system of claim 8, for a decoupled provider at runtime, the management information is used with information provided by a register call to locate a corresponding running process, to connect to the running process, and to locate a subcomponent within the running process that is associated with the management information.

11. (Original) The system of claim 1, the management information includes a probe attribute that is used to indicate that a member of a type is a probe.

12. (Original) The system of claim 11, the type is decorated with a folder attribute.

## UNOFFICIAL COMMUNICATION FOR EXAMINER REVIEW ONLY – PLEASE DO NOT ENTER

13. (Original) The system of claim 1, the management information includes health information that is exposed from an in-process provider.

14. (Original) The system of claim 1, the management information includes health information that indicates health of the application or service.

15. (Original) The system of claim 1, the management information is identified within the attributed application or service using a uniform resource identifier.

16. (Original) The system of claim 1, the management information includes execution information that indicates when the management information should be executed.

17. (Original) The system of claim 1, the management information is exposed from a data source that includes at least one of hardware, software application, and an operating system.

18. (Original) The system of claim 1, the management information includes class definitions that are exposed to a management component.

19. (Original) The system of claim 1, the class definitions are described in a managed object format.

20. (Original) A computer system according to claim 1.

## UNOFFICIAL COMMUNICATION FOR EXAMINER REVIEW ONLY – PLEASE DO NOT ENTER

21. (Original) A computer-readable medium having computer-executable instructions that embodies the system of claim 1.

22. (Previously Presented) A method of managing an application or service, comprising:

receiving the application or service for installation on a system;

attributing selected parts of code of the application or service with management information;

exposing the management information to a management system; and

controlling the application or service based upon the management information that is exposed when the application or service is installed on the system.

23. (Original) The method of claim 22, further comprising generating a manifest of the exposed management information.

24. (Original) The method of claim 22, the management information is exposed from one or more internal processes of a provider.

25. (Original) The method of claim 22, further comprising generating a catalog of all manifests of all available instrumentation data of the system, wherein the catalog is browsed and used to discover a particular instrumentation point.

26. (Original) The method of claim 22, the service is a native service whose code is wrapped with a managed code to facilitate attribution thereof.

27. (Original) The method of claim 22, the attributed code includes at least one of folder and probe attributes.



## UNOFFICIAL COMMUNICATION FOR EXAMINER REVIEW ONLY – PLEASE DO NOT ENTER

28. (Original) The method of claim 22, further comprising authoring the application or service with management information in preparation for a runtime.

29. (Original) The method of claim 22, further comprising generating an instrumentation manifest for the application or service based upon the management information.

30. (Original) The method of claim 29, the instrumentation manifest is stored with a collection of instrumentation manifests that are accessible to a consumer of the management information.

31. (Currently Amended) A system, embodied in a computer-readable medium, for managing an application or service, comprising:

means for attributing selected parts of code of the application or service with health information;

means for exposing the health information in the form of instrumentation definitions;

means for cataloging the instrumentation definitions in a collection of instrumentation definitions; and

means for controlling the application or service based upon the exposed instrumentation definitions when the application or service is installed on the system.

32. (Original) The system of claim 31, further comprising means for identifying the health information with a unique identifier.

## UNOFFICIAL COMMUNICATION FOR EXAMINER REVIEW ONLY – PLEASE DO NOT ENTER

33. (Currently Amended) A computer-readable storage medium having computer-executable instructions for performing a method for managing an application or service, the method comprising:

receiving the application or service for installation on a system;

attributing selected parts of code of the application or service with management information;

exposing the management information to a management system; and

controlling the application or service based upon the management information that is exposed when the application or service is installed on the system.

34. (Currently Amended) A computer-readable storage medium having computer-executable instructions that facilitates a system for managing an application or service, the system comprising:

an application or service for installation on the system; and

an attribution component that facilitates attributing selected parts of code of the application or service with management information;

wherein the system uses the management information to manage the installed application or service.

35. (Original) The computer-readable medium of claim 34, the management information includes a probe attribute that is used to indicate that a member of a type is a probe, which type is decorated with a folder attribute.

36. (Original) The computer-readable medium of claim 34, at runtime, a component of an in-process provider associated with the management information is loaded and invoked, and the management information for a decoupled provider is used with information provided by a register call to locate a corresponding running process, to

UNOFFICIAL COMMUNICATION FOR EXAMINER REVIEW ONLY – PLEASE DO NOT ENTER

connect to the running process, and to locate a subcomponent within the running process that is associated with the management information.

37. (Previously Presented) The system of claim 1, wherein the attributed parts of code are considered probes for use in determining health of the application.